8Д

УДК 681:519.68

В.А. Доценко
Національний технічний університет України «Київський політехнічний інститут»
validator1982@mail.ru

Особливості застосування конструктивного алгоритму зворотного методу для секвенційного числення предикатів

Досліджені області застосування конструктивного алгоритму зворотного методу, здійснена оцінка доцільності застосування конструктивного алгоритму зворотного методу, виходячи з поставлених задач. Проведена оцінка складності побудови початкової множини сприятливих наборів. Розглянуто приклади застосування конструктивного алгоритму зворотного методу для секвенцій різного вигляду, а також випадку, коли застосування конструктивного алгоритму зворотного методу не є ефективним.

Вступ

Конструктивний алгоритм зворотного методу (КАЗМ) був нами розглянутий у працях [1-3]. Він є розширенням зворотного методу (ЗМ) С.Ю. Маслова [4] за рахунок вдосконалення його та застосування евристик із упорядкування виведення та зі скорочення перебору.

Перш ніж застосовувати конструктивний алгоритм зворотного методу для пошуку виведення, слід знайти відповіді на наступні питання:

1. Якою є предметна область застосування КАЗМ?
2. Яка область доцільності застосування КАЗМ?
3. Як можна оцінити складність КАЗМ?
4. Чи можливі випадки неефективної роботи КАЗМ?

Під першим питанням ми розуміємо числення, для яких може бути застосований КАЗМ. Під другим питанням ми розуміємо задачі та формат вхідних формул, для вирішення яких доцільно застосовувати КАЗМ. Під третім питанням ми розуміємо потребу здійснення оцінки як кількості кроків на застосування алгоритму, так і для порівняльного аналізу складності застосування алгоритму для різних вхідних формул. Під четвертим питанням ми розуміємо пошук випадків, для яких процес виведення за КАЗМ потребує істотно більшої кількості кроків, ніж для іншого варіанта застосування КАЗМ. Якщо такі випадки можливі, необхідно визначити шляхи щодо мінімізації появи таких випадків.

В даній статті ми дамо відповіді на поставлені питання, а також наведемо приклади роботи КАЗМ у секвенційному численні предикатів першого порядку.

Область застосування КАЗМ

Конструктивний алгоритм зворотного методу є об'єднанням методів скорочення перебору та упорядкування процесу виведення при застосуванні зворотного методу. Нами було доведено, що їх застосування не призводить до втрати методом виводимості. Тому область застосування КАЗМ збігається з такою для ЗМ. На даний момент існують формулювання ЗМ для логіки висловлювань [5], теорії предикатів

першого порядку з рівністю [6] та без [7], для модальної логіки S4 [8], для лінійної логіки [9]. Щодо інших логічних числень, які не були нами перелічені, то для них застосування ЗМ (а отже, і КАЗМ) можливе – для цього потрібно лише провести аналіз допустимості та здійснити формулювання методу з урахуванням синтаксису та семантики даного числення. На даний момент нами було сформульовано КАЗМ для логіки висловлювань та для логіки предикатів першого порядку без рівності. Відмінністю КАЗМ для різних числень є входження до КАЗМ евристичних доповнень, що враховують специфіку конкретного числення, а отже, можуть бути застосовані лише на ньому.

Формат вхідних формул, з якими працює КАЗМ, є ширшим від такого для методу *резолюцій*. Якщо розглядати секвенційне числення предикатів першого порядку, то КАЗМ здатен опрацьовувати формули вигляду:

$$\Gamma \Rightarrow \Delta , \qquad (1)$$

де $\Gamma = (\forall x)(\bigcap_{i=1}^{n} A_i)$, $n > 0$, $\boldsymbol{x}$ – список змінних, $A_i = \bigcup_{s=1}^{\delta_i} A_{i_s}$, $\delta_i \ge 0$, $\Delta = (\exists y)(\bigcup_{j=1}^{m} B_j)$, $m > 0$, $\boldsymbol{y}$ – список змінних, $B_j = \bigcap_{k=1}^{\delta_j} B_{j_k}$, $\delta_j \ge 0$, A_{p_l}, B_{u_v} – відкриті формули, що можуть складатися з одної літери. Графічно рівні літери скорочені до однієї.

Тут Γ, Δ знаходяться у зведеній нормальній формі.

Також КАЗМ опрацьовує формули вигляду $\Rightarrow \Delta$ (2) та $\Gamma \Rightarrow$ (3).

КАЗМ будує дерево виведення без початкового перенесення усіх підформул до антицидента чи до сукцидента.

Область доцільності застосування КАЗМ

Ми приведемо ключові переваги КАЗМ та окреслимо задачі, для яких ці переваги є ключовими при виборі алгоритму пошуку виведення. Порівнюватимемо КАЗМ з методом резолюції. Випишемо переваги кожного із цих методів у порівнянні їх між собою.

Переваги КАЗМ:

– вхідна формула може бути представлена секвенціями вигляду (1-3);
– вхідну формулу не є необхідним перетворювати до ССФ, достатньо перетворити її до зведеної нормальної форми;
– дерево виведення може бути просто трансльоване на природну мову, кожен з вузлів дерева є аксіомою числення, в якому здійснюється виведення;
– відсутня необхідність перетворення вхідної формули до вигляду (3), де формула представлена у ССФ.

Переваги методу резолюції:

– відсутня трудомістка процедура побудови початкової множини сприятливих наборів;
– наявні для модифікацій методу резолюцій евристики є більш простими за такі у КАЗМ (а отже, потребують меншої кількості операцій);
– відсутня необхідність повторної побудови дерева виведення з початку, у випадку невірного вибору поточного сприятливого набору.

Виходячи з приведеної інформації, ми вважаємо, що для узагальненого випадку, коли стоїть лише задача встановлення виводимості формули, а вхідні формули представлені у вигляді (3) або ж відносно просто до нього перетворюються, доцільнішим є застосовувати метод резолюції. Якщо перетворення формул до того вигляду, у якому їх здатен обробляти метод резолюції, є трудомістким, то визна-

чення кращого з цих двох методів не є однозначним. Якщо є потреба у трансляції кожного вузла дерева виведення на природну мову (така необхідність є в експертних системах для перевірки експертом ходу розмірковування або видачі його користувачу), то слід застосовувати зворотний метод.

Оцінка складності КАЗМ

Перш за все відмітимо, що КАЗМ відноситься до методів повного перебору, а отже, має експоненційну складність. Щодо можливої оцінки (порівняльної оцінки) його складності, то КАЗМ має свої особливості, які безпосередньо впливають на його складність.

У кожного методу є свої переваги і недоліки, при цьому, як правило, недоліки є продовженням переваг. Це стосується і зворотного методу. Основною його ідеєю є побудова дерева виводу з аксіом (сприятливих наборів), що будуються зі структури формули, яку необхідно довести. Проте це породжує проблему – побудованих сприятливих наборів (СН) значно більше, ніж це є необхідно для побудови дерева виведення. Для того, щоб зменшити навантаження на алгоритм, у попередніх працях нами було приведено ряд евристик та допоміжних правил, направлених на скорочення потужності множини сприятливих наборів.

Ми здійснимо оцінку потужності множини початкових сприятливих наборів. Така оцінка буде проводитись виходячи зі структури формули, яку нам необхідно довести. Отримана формула й буде орієнтовною оцінкою складності методу, оскільки вивести узагальнену формулу розрахунку кількості кроків, які витрачає алгоритм на виведення формули, не є можливим – в залежності від вхідної формули, від шляху, за яким піде алгоритм, кількість кроків алгоритму може бути істотно різною, і здійснити попередньо таку оцінку для КАЗМ не є можливим. Щодо оцінки кількості сприятливих наборів у початковій множині сприятливих наборів, то від них в першу чергу залежить величина дерева виведення – чим більша кількість сприятливих наборів, тим більшим буде дерево пошуку виведення. Другим фактором є непередбачувана специфіка кожної вхідної формули.

Отож здійснимо оцінку потужності початкової множини сприятливих наборів.

Кількістю додатних літер атому a ми будемо називати число літер цього атому, що входять у початкову формулу без заперечення. Це число позначатимемо n_a^+.

Кількістю від'ємних літер атому a ми будемо називати число літер цього атому, що входять у початкову формулу із запереченням. Це число позначатимемо n_a^-.

Кількістю комбінацій атому a ми будемо називати добуток кількості додатних літер атому a і кількості від'ємних літер атому a. Це число позначатимемо κ_a.

Тепер ми можемо сформулювати нашу оцінку:

Нехай F *– формула, в якій графічно рівні літери скорочені до однієї. Тоді кількість сприятливих наборів, що можуть бути побудовані з цієї формули, складатиме*

$$\mathrm{K} = \sum_{i=1}^{p} \kappa_i , \qquad (4)$$

де p *– кількість атомів формули* F.

Приклад 1. Нехай початковою формулою буде

$$F = A \wedge B \wedge C \vee \neg A \wedge B \wedge C \vee A \wedge \neg B \vee \neg C \wedge D \vee \neg B \wedge D \vee \neg D .$$

Тоді $n_A^+ = 2$, $n_A^- = 1$, $\kappa_A = 2$, $n_B^+ = 2$, $n_B^- = 2$, $\kappa_B = 4$,

$n_C^+ = 2$, $n_C^- = 1$, $\kappa_C = 2$, $n_D^+ = 2$, $n_D^- = 1$, $\kappa_D = 2$.

Застосовуючи формулу (4), маємо: $\mathrm{K} = 2 + 4 + 2 + 2 = 10$.

Побудуємо початкову множину сприятливих наборів для F :

$1_1 2_1,\ \ 1_2 3_2,\ \ 1_2 5_1,\ \ 1_3 4_1,\ \ 2_1 3_1,\ \ 2_2 3_2,\ \ 2_2 5_2,\ \ 2_3 4_1,\ \ 4_2 6_0,\ \ 5_2 6_0$.

Як бачимо, кількість початкових сприятливих наборів становить 10, що збіглося з нашою оцінкою.

Приклади застосування КАЗМ для секвенційного числення предикатів першого порядку

Розглянемо тепер декілька прикладів застосування КАЗМ пошуку виведення вхідної формули у численні предикатів першого порядку, які демонструють найбільш характерні особливості побудови ним дерева виведення. Перші два приклади показують застосування роботу КАЗМ у секвенційному численні предикатів першого порядку для вхідних формул, представлених у вигляді секвенцій різного виду, третій приклад демонструє випадок неефективної роботи КАЗМ, коли під час побудови гілки дерева виведення доводиться повертатись до самого початку, вибирати поточним інший номер з індексом і заново будувати дерево виведення.

Розглянемо спершу приклад, який показує застосування КАЗМ для вхідної формули, представленої у вигляді (2).

Приклад 2. Нехай маємо секвенцію вигляду

$$\Rightarrow \exists x \exists y (P(x,y) \wedge R(f(x)) \vee \neg P(f(x), f(y))) \vee \exists x \exists y (\neg P(x, f(x)) \wedge \neg R(y) \wedge P(f(x), f(x))).$$

Застосуємо до неї описаний нами вище алгоритм. Маємо

1. Перейменування змінних:

$$\Rightarrow \exists x_1 \exists x_2 (P(x_1, x_2) \wedge R(f(x_1)) \vee \neg P(f(x_1), f(x_2))) \vee$$
$$\vee \exists x_3 \exists x_4 (\neg P(x_3, f(x_4)) \wedge \neg R(x_4) \wedge P(f(x_3), f(x_3))) .$$

2. Внесення мета змінних:

$$\Rightarrow P(t_1,t_2) * \{t_1/x_1, t_2/x_2\} \wedge R(f(t_1)) * \{t_1/x_1\} \vee \neg P(f(t_1), f(t_2)) * \{t_1/x_1, t_2/x_2\} \vee$$
$$\vee \neg P(t_3, f(t_4)) * \{t_3/x_2\} \wedge \neg R(t_4) * \{t_4/x_4\} \wedge P(f(t_3), f(t_3)) * \{t_3/x_3\} .$$

3. Побудова початкової МСН:

$$\{[(P(t_5, f(t_5)), \{t_5/t_1, t_1/x_1, f(t_5)/t_2, t_2/x_2\}); (\neg P(t_5, f(t_5)), \{t_5/t_3, t_3/x_3\})],$$
$$[(R(f(t_6)), \{t_6/t_1, t_1/x_1\}); (\neg R(f(t_6)), \{f(t_6)/t_4, t_4/x_4\})],$$
$$[(\neg P(f(t_7), f(t_7)), \{t_7/t_1, t_1/x_1, t_7/t_2, t_2/x_2\}); (P(f(t_7), f(t_7)), \{t_7/t_3, t_3/x_3\})]\}.$$

В номерах ця МСН має вигляд:

$$\{[(1_1, \xi_1); (3_1, \xi_2)], [(1_2, \xi_3); (3_2, \xi_4)], [(2_0, \xi_5); (3_3, \xi_6)]\},$$

де ξ_i , $i = 1, \ldots, 6$ – відповідні F-підстановки.

4. Побудова дерева виведення

4.1. $\sigma_1 = \{t_8/t_5\}$, $\sigma_2 = \{t_8/t_6\}$, $\sigma = \{t_8/t_5, t_5/t_1, f(t_5)/t_1, t_2/x_2, t_8/t_6, t_6/t_1, t_1/x_1\}$

$1_1 3_1$ $\quad$ $1_2 3_2$

$3_1 3_2\,(1)$

Поточний СН має вигляд:

$$[(\neg P(t_8, f(t_8)), \{t_8/t_5, t_5/t_3, t_3/x_3\}); (\neg R(f(t_8)), \{t_8/t_6, f(t_6)/t_4, t_4/x_4\})].$$

4.2. $\sigma_1 = \{t_9/t_8\}$, $\sigma_2 = \{t_9/t_7\}$,

$$\sigma = \{t_9/t_8, t_8/t_5, t_5/t_3, t_3/x_3, t_8/t_6, f(t_6)/t_4, t_4/x_4, t_9/t_7, t_7/t_3, t_3/x_3\} .$$

4.3. $\sigma = \{t_9/t_7, t_7/t_1, t_1/x_1, t_7/t_2, t_2/x_2\}$

$$2_0 3_3 \qquad 3_1 3_2(1)$$
$$2_0(1,3)$$
$$|$$
$$\gamma(1,2,3)$$

Тут γ – символ порожнього сприятливого набору.

5. Виведення завершене виводом секвенції:

$$\Rightarrow P(t_8, f(t_8)) \wedge R(f(t_8)) * \{t_8/t_5, t_5/t_1, t_1/x_1, f(t_5)/t_2, t_2/x_2, t_8/t_6, t_6/t_1, t_1/x_1\} \vee$$
$$\vee \neg P(f(t_9), f(t_9)) * \{t_9/t_7, t_7/t_1, t_1/x_1, t_7/t_2, t_2/x_2\} \vee$$
$$\vee (\neg P(t_9, f(t_9)) \wedge \neg R(f(t_9)) \wedge P(f(t_9), f(t_9))) * \{t_9/t_8, t_8/t_6, f(t_6)/t_4, t_4/x_4, t_9/t_7, t_3/x_3\}.$$

Якщо ми спростимо підстановки, то отримаємо:

$$\Rightarrow P(t_8, f(t_8)) \wedge R(f(t_8)) * \{t_8/x_1, f(t_8)/x_2\} \vee \neg P(f(t_9), f(t_9)) * \{t_9/x_1, t_7/x_2\} \vee$$
$$\vee (\neg P(t_9, f(t_9)) \wedge \neg R(f(t_9)) \wedge P(f(t_9), f(t_9))) * \{f(t_9)/x_4, t_9/x_3\}.$$

Ми встановили виводимість початкової секвенції.

Розглянемо тепер приклад встановлення виводимості $\alpha\beta$-секвенції (1).

Приклад 3. Нехай маємо секвенцію вигляду

$$\forall x \forall y(\neg P(x,y) \vee \neg R(f(x)) \wedge P(f(x), f(y))) \Rightarrow \exists x \exists y(\neg P(x, f(x)) \wedge \neg R(y) \wedge P(f(x), f(x))).$$

Застосовуємо алгоритм.

1. Перейменування змінних:

$$\forall x_1 \forall x_2(\neg P(x_1,x_2) \vee \neg R(f(x_1)) \wedge P(f(x_1), f(x_2))) \Rightarrow$$
$$\Rightarrow \exists x_3 \exists x_4(\neg P(x_3, f(x_3)) \wedge \neg R(x_4) \wedge P(f(x_3), f(x_3))).$$

2. Внесення мета змінних:

$$(\neg P(t_1,t_2) * \{t_1/x_1, t_2/x_2\} \vee \neg R(f(t_1)) * \{t_1/x_1\}) \wedge P(f(t_1), f(t_2)) * \{t_1/x_1, t_2/x_2\} \Rightarrow$$
$$\vee \neg P(t_3, f(t_4)) * \{t_3/x_2\} \wedge \neg R(t_4) * \{t_4/x_4\} \wedge P(f(t_3), f(t_3)) * \{t_3/x_3\}.$$

3. Побудова початкової МСН:

$$\{[(\neg P(t_5, f(t_5)), \{t_5/t_1, t_1/x_1, f(t_5)/t_2, t_2/x_2\})^\alpha; (\neg P(t_5, f(t_5)), \{t_5/t_3, t_3/x_3\})^\beta],$$
$$[(\neg R(f(t_6)), \{t_6/t_1, t_1/x_1\})^\alpha; (\neg R(f(t_6)), \{f(t_6)/t_4, t_4/x_4\})^\beta],$$
$$[(P(f(t_7), f(t_7)), \{t_7/t_1, t_1/x_1, t_7/t_2, t_2/x_2\})^\alpha; (P(f(t_7), f(t_7)), \{t_7/t_3, t_3/x_3\})^\beta]\}.$$

В номерах ця МСН має вигляд: $\{[(1_1^\alpha, \xi_1); (3_1^\beta, \xi_2)], [(1_2^\alpha, \xi_3); (3_2^\beta, \xi_4)], [(2_0^\alpha, \xi_5); (3_3^\beta, \xi_6)]\}$,
де ξ_i, $i = 1,\ldots,6$ – відповідні F-підстановки.

4. Побудова дерева виведення:

$$[(1_1^\alpha, \{t_5/t_1, t_1/x_1, f(t_5)/t_2, t_2/x_2\}); (3_1^\beta, \{t_5/t_3, t_3/x_3\})]$$
$$[(1_2^\alpha, \{t_6/t_1, t_1/x_1\}); (3_2^\beta, \{f(t_6)/t_4, t_4/x_4\})]$$
$$[(3_1, \{t_8/t_5, t_5/t_3, t_3/x_3\})^\beta; (3_2, \{t_8/t_6, f(t_6)/t_4, t_4/x_4\})^\beta] \quad (1^\alpha * \theta_1)$$
$$[(2_0^\alpha, \{t_7/t_1, t_1/x_1, t_7/t_2, t_2/x_2\}); (3_3^\beta, \{t_7/t_3, t_3/x_3\})]$$
$$[(2_0^\alpha, \{t_7/t_1, t_1/x_1, t_7/t_2, t_2/x_2\})] \quad (1^\alpha * \theta_1, 3^\beta * \theta_3)$$
$$|$$
$$\gamma \quad (1^\alpha * \theta_1, 2^\alpha * \theta_2, 3^\beta * \theta_3)$$

де $\theta_1 = \{t_8 / t_5, t_5 / t_1, f(t_5) / t_1, t_2 / x_2, t_8 / t_6, t_6 / t_1, t_1 / x_1\}$, $\theta_2 = \{t_9 / t_7, t_7 / t_1, t_1 / x_1, t_7 / t_2, t_2 / x_2\}$ $\theta_3 = \{t_9 / t_8, t_8 / t_6, f(t_6) / t_4, t_4 / x_4, t_9 / t_7, t_8 / t_5, t_7 / t_3, t_5 / t_3, t_3 / x_3\}$.

5. Виведення завершене виводом секвенції (підстановки скоротили):

$$(P(t_8, f(t_8)) \vee \neg R(f(t_8)) * \{t_8 / x_1, f(t_5) / x_2\}) \wedge \neg P(f(t_9), f(t_9)) * \{t_9 / x_1, t_9 / x_2\} \Rightarrow$$
$$(\neg P(t_9, f(t_9)) \wedge \neg R(f(t_9)) \wedge P(f(t_9), f(t_9))) * \{f(t_9) / x_4, t_9 / x_3\}.$$

Приведений приклад демонструє лінійний порядок виведення. Проте не завжди таке може бути здійснене. Розглянемо приклад, в якому процес виведення є неефективним.

Приклад 4. Розглянемо приклад «цікаве життя», який був запозичений з [11].

Задані твердження:

1. Усі небідні та розумні люди щасливі: $\exists x(\neg poor(x) \wedge smart(x) \to happy(x))$.
2. Людина, яка читає книги, розумна: $\forall y(read(y) \to smart(y))$.
3. Джон вміє читати і є багатою людиною: $read(John) \wedge \neg poor(John)$.
4. Щасливі люди живуть цікавим життям: $\forall z(happy(z) \to exiting(z))$.

Необхідно отримати відповідь на питання: Чи існує людина, яка живе цікавим життям? Питання має вигляд: $\exists w(exiting(w))$.

Запишемо формулу, яку необхідно вивести для відповіді на поставлене питання. Вона має вигляд:

$$(\exists x(\neg poor(x) \wedge smart(x) \to happy(x))) \wedge (\forall y(read(y) \to smart(y))) \wedge$$
$$(read(John) \wedge \neg poor(John)) \wedge (\forall z(happy(z) \to exiting(z))) \Rightarrow \exists w(exiting(w)).$$

Перейменуємо предикати на літери A та B.

$$(\exists x(\neg A_{11}(x) \wedge A_{12}(x) \to A_{13}(x))) \wedge (\forall y(A_{21}(y) \to A_{22}(y))) \wedge (A_{31}(J) \wedge \neg A_{32}(J)) \wedge$$
$$(\forall z(A_{41}(z) \to A_{42}(z))) \Rightarrow \exists w(B_{10}(w)).$$

Приведемо формулу до зведеної нормальної форми і введемо метазмінні.

$$(poor(f(t_1, t_2)) \vee \neg smart(f(t_1, t_2)) \vee happy(f(t_1, t_2))) * \{f(t_1, t_2) / x\} \wedge$$
$$(\neg read(t_1) \vee smart(t_1)) * \{t_1 / y\} \wedge$$
$$(read(J) \wedge \neg poor(J)) \wedge (\neg happy(t_2) \vee exiting(t_2)) * \{t_2 / z\} \Rightarrow exiting(t_3) * \{t_3 / w\}.$$

Перейдемо до позначення предикатів номерами і сформуємо початкову множину сприятливих наборів. Для зручності при кожному номері будемо писати два індекси – перший індекс буде позначати номер літери в підформулі, другий індекс (через знак «/») позначатиме кількість літер у даній підформулі. Таким чином, не буде потреби дивитись на початкову формулу, щоб дізнатись кількість літер у підформулі.

$$[(1_{1/3}, \{J / f(t_1, t_2), f(t_1, t_2) / x\})^{\alpha}; (3_{2/2}, \{\varnothing\})^{\alpha}],$$
$$[(1_{2/3}, \{t_4 / t_1, t_5 / t_2, f(t_1, t_2) / x\})^{\alpha}; (2_{2/2}, \{f(t_4, t_5) / t_1, t_1 / y\})^{\beta}]$$
$$[(1_{3/3}, \{t_6 / t_1, t_7 / t_2, f(t_1, t_2) / x\})^{\alpha}; (4_{1/2}, \{f(t_6, t_7) / t_2, t_2 / z\})^{\alpha}]$$
$$[(2_{1/2}, \{J / t_1, t_1 / y\})^{\alpha}; (3_{1/2}, \{\varnothing\})^{\alpha}], \ [(4_{2/2}, \{t_7 / t_2, t_2 / z\})^{\alpha}; (5_0, \{t_7 / t_3, t_3 / w\})^{\beta}].$$

Застосовуючи алгоритм прополки, переносимо число 5 в залежність п'ятого сприятливого набору. Вибираємо згідно з правилом пріоритету поточний сприятливий набір та починаємо будувати дерево виведення.

$$[(1_{1/3}, \{J / f(t_1, t_2), f(t_1, t_2) / x\})^{\alpha}; (3_{2/2}, \{\varnothing\})^{\alpha}]$$
$$[(1_{2/3}, \{t_4 / t_1, t_5 / t_2, f(t_1, t_2) / x\})^{\alpha}; (2_{2/2}, \{f(t_4, t_5) / t_1, t_1 / y\})^{\beta}]$$
$$[(1_{3/3}, \{t_6 / t_1, t_7 / t_2, f(t_1, t_2) / x\})^{\alpha}; (4_{1/2}, \{f(t_6, t_7) / t_2, t_2 / z\})^{\alpha}]$$
$$[(3_{2/2}, \{\varnothing\})^{\alpha}; (2_{2/2}, \{J / f(t_4, t_5), f(t_4, t_5) / t_1, t_1 / y\})^{\alpha}; (4_{1/2}, \{J / f(t_6, t_7), f(t_6, t_7) / t_2, t_2 / z\})^{\alpha}](1^{\alpha})$$

Подальша побудова дерева виведення блокується за правилом VI – поточна множина сприятливих наборів містить лише такі СН, застосування до яких правила Б призведе до отримання повного набору індексів більш ніж по одному номеру. Для продовження виведення необхідно повернутись до початкової множини сприятливих наборів і розпочати гілку дерева виведення з іншого номеру. При цьому помітимо номер 1 як такий, пошук номерів з індексами по якому слід вести в останню чергу.

$$[(2_{1/2},\{J/t_1,t_1/y\})^{\alpha};(3_{1/2},\{\varnothing\})^{\alpha}]$$

$$[(1_{2/3},\{t_4/t_1,t_5/t_2,f(t_1,t_2)/x\})^{\alpha};(2_{2/2},\{f(t_4,t_5)/t_1,t_1/y\})^{\beta}]$$

$$[(1_{2/3},\{J/f(t_4,t_5),t_4/t_1,t_5/t_2,f(t_1,t_2)/x\})^{\alpha};(3_{1/2},\{\varnothing\})^{\alpha}](2^{\alpha}*\theta_1)$$

$$[(1_{1/3},\{J/f(t_1,t_2),f(t_1,t_2)/x\})^{\alpha};(3_{2/2},\{\varnothing\})^{\alpha}]$$

$$[(1_{2/3},\{J/f(t_4,t_5),t_4/t_1,t_5/t_2,f(t_1,t_2)/x\})^{\alpha};(1_{1/3},\{J/f(t_1,t_2),f(t_1,t_2)/x\})^{\alpha}](2^{\alpha}*\theta_1,3^{\alpha}*\theta_2)$$

$$[(1_{3/3},\{t_6/t_1,t_7/t_2,f(t_1,t_2)/x\})^{\alpha};(4_{1/2},\{f(t_6,t_7)/t_2,t_2/z\})^{\alpha}]$$

$$[(4_{1/2},\{J/f(t_6,t_7),f(t_6,t_7)/t_2,t_2/z\})^{\alpha}](1^{\alpha}*\theta_1,2^{\alpha}*\theta_2,3^{\alpha}*\theta_3)$$

$$[(4_{2/2},\{t_7/t_2,t_2/z\})^{\alpha}](5^{\beta}*\theta_5)$$

$$\gamma(1^{\alpha}*\theta_1,2^{\alpha}*\theta_2,3^{\alpha}*\theta_3,4^{\alpha}*\theta_4,5^{\beta}*\theta_5)$$

де $\theta_1=\{J/f(t_1,t_2),J/f(t_4,t_5),t_4/t_1,t_5/t_2,J/f(t_6,t_7),t_6/t_1,t_7/t_2,f(t_1,t_2)/x\}$,

$\theta_2=\{J/t_1,t_1/y,t_4/t_1,t_5/t_2,f(t_1,t_2)/x\}$,

$\theta_3=\{\varnothing\}$,

$\theta_4=\{J/f(t_6,t_7),f(t_6,t_7)/t_2,t_2/z,t_7/t_2,t_2/z\}$,

$\theta_5=\{J/t_1,t_1/y,t_4/t_1,t_5/t_2,f(t_1,t_2)/x\}$.

Виведення завершене виводом порожнього сприятливого набору, а отже, початкова теорема доведена (твердження, що існує людина, яка живе цікавим життям, виведене).

Як було продемонстровано на наведеному прикладі, при аналізі кількості літер у підформулах (індексів при номерах) при залученні сприятливих наборів для застосування правила Б слід віддавати перевагу в першу чергу тим номерам, по яким є менша кількість індексів (підформулам з меншою кількістю літер). В такому випадку отриманий після застосування правила Б сприятливий набір буде містити меншу кількість номерів з індексами, а отже, ймовірність появи випадку, коли спрацьовує правило VI, буде знижена. При застосуванні такої методики (застосування приведеного правила до формули з прикладу 2) виведення одразу буде вестись за другою гілкою.

Висновок

КАЗМ може застосовуватись для широкого класу формул, але область доцільності його застосування є меншою. В даній статті вказано задачі, для яких застосування КАЗМ є доцільнішим за застосування методу резолюції. Ми також навели приклади застосування КАЗМ для виведення формул у секвенційному численні предикатів першого порядку, показали випадок, коли виведення КАЗМ є неефективним, та вказали методику для зменшення таких випадків.

Література

1. Доценко В.А. Конструктивний алгоритм зворотного методу для числення висловлювань // Системні дослідження та інформаційні технології. – 2007. – № 2. – С. 59-73.
2. Доценко В.А. Конструктивний алгоритм зворотного методу для секвенцій загального вигляду // Искусственный интеллект. – 2006. – № 3. – С. 120-128.
3. Доценко В.А. Конструктивний алгоритм зворотного методу для секвенційного числення предикатів // Вісник національного технічного університету України «Київський політехнічний інститут»: інформатика, управління та обчислювальна техніка. – 2007. – № 46. – С. 211-225.
4. Маслов С.Ю. Обратный метод установления выводимости в классическом исчислении предикатов // ДАН ССР. – 1964. – № 1. – С. 17-20.
5. Катречко С.Л. Модификации обратного метода С.Ю. Маслова // Материалы X Всесоюзной конференции по логике, методологии и философии науки. – М., 1990.
6. Voronkov A. Theorem proving in non-standard logics based on the inverse method. In D. Kapur, editor // Proceedings of the 11-th International Conference on Automated Deduction, Saratoga Springs. – New York, 1992. Springer-Verlag LNCS 607. – P. 618-662.
7. Degtyarev A., Voronkov A. Equality Elimination for the Inverse Method and Extension Procedures. – Uppsala (Sweden), 1995.
8. Degtyarev A., Voronkov A. The inverse Method, in A. Robinson & A. Voronkov eds. Handbook of Automated Reasoning, 2001. – P. 181-270.
9. Chaudhuri K. Theorem Proving with the Inverse Method for Linear Logic. – Pittsburgh PA: Carnegie Mellon University, 2004.
10. Birstunas A., Norgela S. Inverse method for modal logic S4 // Matemetikos ir informatikos institutas, T34, 2003.
11. Люгер Д. Искусственный интеллект. – М.: Мир, 2003. – 690 с.

В.А. Доценко
Особенности применения конструктивного алгоритма обратного метода для секвенциального исчисления предикатов
Исследована область применения конструктивного алгоритма обратного метода исходя из поставленных задач. Проведена оценка сложности построения начального множества благоприятных наборов. Рассмотрены примеры применения конструктивного алгоритма обратного метода для секвенций различного вида, а также случая, когда применение конструктивного алгоритма обратного метода не является эффективным.

Стаття надійшла до редакції 24.07.2008.